

AI/HPC時代を実現するためのソフトウェア 「HPC向けコンテナSingularity開発動向」

2022年12月5日 富士ソフト アキバプラザ

Pacific Teck Japan Senior Engineer 森本 賢治

Pacific Teck Japan ポートフォリオ

- ジョブスケジューラ



- HPC向けコンテナ



- ストレージソフトウェア

- 並列ファイルシステム



- オブジェクトストレージ



- S3互換クラウドストレージ



- ルール指向高機能ストレージ構築ミドルウェア



- クラスタ構築・管理ツール



- 開発及びプロファイリングツール



随時拡大していきます



SingularityCE

- コミュニティ主導のオープンソース。有償サポートなし。
- 新機能は最初に実装されるが、セキュリティ対応や修正は最新版にのみ適用。



SingularityPRO

- Sylabs社による検証・署名済みバイナリパッケージを提供。
- 長期サポート(リリースから2年)・機能追加・プラグイン開発を依頼可能。
- 脆弱性の公開前に対応済みバイナリが提供され、バックポートにも対応。



Singularity Enterprise

- コンテナランタイムとしてはSingularityPROを提供。
- イメージ作成のオンデマンド環境RemoteBuilder, 電子署名の公開鍵管理KeyStore, イメージのリポジトリLibraryの3機能をKubernetes上で実現。EKS, GKEもサポート。ただし、Kubernetesの運用はユーザーマター。

SingularityとApptainerの関係 —総括—



Sylabs

- 元々、主要開発者は殆どがSylabs所属、技術の根幹を成すSIF規格（ファイルフォーマット及びAPI）もSylabsがメンテナンスを継続。
- HPCngへのリソース提供、リリースコントロールもSylabsが担っていたが、方向性で折り合いつかず、プロジェクトをCEとしてフォークした。
- Sylabsとしては、Singularityからの派生ソフトウェアについては、SIFに準拠している限りは互換性を排除しない。



Sylabsからの支援を失ってプロジェクトごとLinuxFoundationへ移管。その際Apptainerに改称。公開されるSingularityCEからセキュリティ対応及びバグ修正の多くを取り込んでいるが、Sylabs依存する仕様の削除が進んだ模様。

Singularityは、
コンテナランタイムのみを
指すのではなく・・・
コンテナイメージの規格としての
SIF形式を含む概念。

一般的にSingularityの利点
と言われる特徴の中には、
SIFの優位性が多数ある。



Singularityコンテナランタイムが、
SIF以外のイメージからコンテナを
他のコンテナランタイムが、SIFから
コンテナを起動してもよい。

標準規格OCIへの対応を進める。

コンテナランタイム

- コンテナを起動、その中でアプリケーションを動かすソフトウェア。
- High/Low level の2階層で構成される。
 - OCI寄り実装
 - High Level : Docker, Containerd, K8s, podman
 - Low Level : runc, runnc, gVisor, Kata Container
 - Singularity: High/Low 同居。HPC特化で独自に進化

コンテナイメージ

- アプリケーションとそれに必要なランタイムを含んだもの。
- 特定のイメージフォーマット(形式)に従って保存される
 - Docker Image manifest V2 Schema2
 - Docker Image をベースに標準規格として策定された、OCI (Open Container Initiative) Image 形式
 - Singularityの標準、イメージを1ファイルにカプセルする SIF(Singularity Image Format)

SIF利用による優位性

可搬性

- シングルファイルで保存するため、持ち運びが用意。
- 通常のNASでの管理も可能。
- 電子署名による同一性を保証。

フットプリント

- 内部はSquashFS。tar+gzipを直接マウントしているイメージ。
- オンメモリでアクセスでき、事前に展開する必要がない。メモリやCPU負荷も極めて小さい。

パフォーマンス

- ファイルアクセスはローカルのオンメモリで行われ、共有ファイルシステムへの、メタデータアクセス負荷を極小化できる。通信負荷も下げられる。

特に昨今、AIのワークロードがPythonベースで実行されるため、様々な環境がユーザーのホームディレクトリに展開され、大きな問題となっている。

「共有ファイルシステムは常に高負荷。
ありがとうPython。
もうAnacondaは禁止したい。」

Dr. Stanzione @ BeeGFS BoF SC22



参考: 弊社ブログ

[Singularity利用による共有ファイルシステムの負荷軽減](#)

標準規格のOCI対応を目指す背景

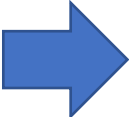


Singularity

- コンテナをホスト環境の一部として捉え、各リソースを共有して使えることを目指した。
- シェルからの**HPCアプリケーションの起動**に特化した使い勝手。
- イメージをシングルファイル内で固定して再現性を担保。

OCI(Docker)

- ホストとコンテナを別システムとして隔離することを求めた。特にマルチテナント環境では必須要件。
- 複数コンテナを組み合わせ、特定用途向けサービスの**システム構築**に用いられることが多い。
- レイヤーを重ねることで、差分アップデートを容易にしたイメージ配布形態。

- 
- イメージの更新や世代管理は容易だが、高性能を得にくいDockerでAIワークロードを動かすことが標準的になった結果、それらをHPC環境で動かすことが求められた。
 - その一方、オーケストレーション環境でHPCアプリケーションを使ったり、逆にSIFの高速性を活かしてAIワークロードを運用したい等、相互利用を意識したりする必要が出て来た。

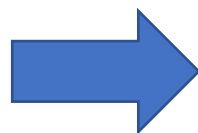
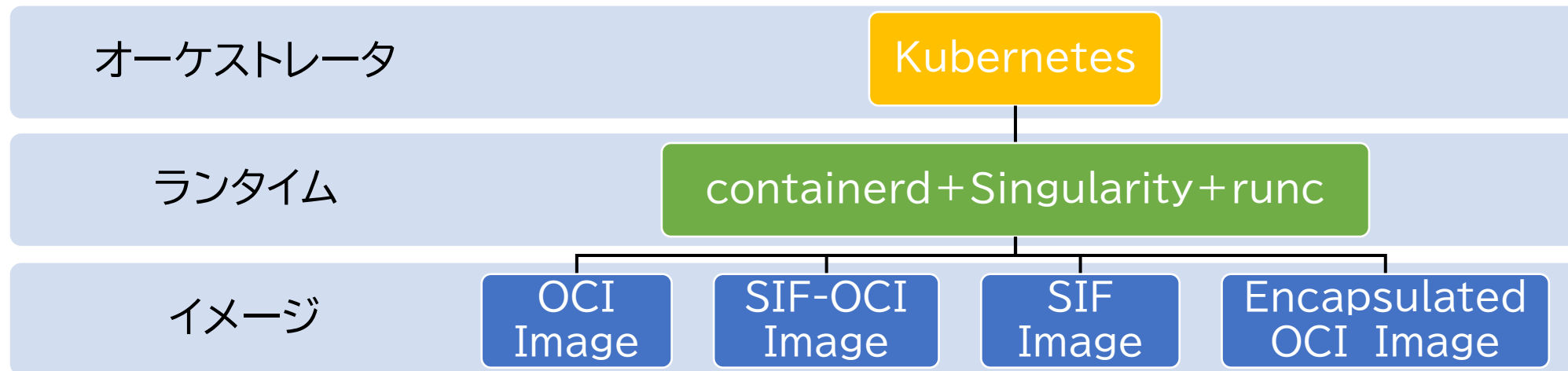
OCI規格を通じた他実装との連携のイメージ

Singularityには
オーケストレータがない

OCIにはKubernetesという
強力なデファクトスタンダード

SIFの利用はパフォーマンス面で
高い優位性がある。

(サンプルイメージ:例えばこのようなものが実現できるかもしれない)



Singularity本来の利便性や使い方はそのままに、
コンテナの業界標準に自然に組み込まれることで、
分野や用途を問わず、相互にメリットを享受。

コンテナの抱えるセキュリティ課題とSingularityの対応

透明性の確立

- コンテナに脆弱性や問題のあるアプリケーションが含まれてしまうと発見が困難。
- SBOMにより、コンテナがどのようなソフトウェアで構成されているのか把握できるようにする。
- 米政府機関では、この対策がないブラックボックス状態のイメージの実行を禁じる動き。
- SIFはシングルファイルでイメージを構成するため、ファイル差し替えや選択ミスを防ぐ仕組みが必要。

電子署名の正当性

- OpenPGPのクリアテキスト形式がイメージ署名に用いられてきたが、鍵の正当性を保証する方法がない。
- PEM形式を使う方法もあるが、既に実績が多いDSSE形式をCEで検証開始。

リソース制限の有効性

- cgroups v2やseccomp等により、コンテナが無制限にリソースにアクセスできない仕組みの実装と運用。

特権の排除

- rootlessやunprivilegedというキーワード。root権限を全く必要としないコンテナ利用の一般化。
- 実現には様々な方法があり、CEをベースに模索。(fakeroot, proot等のサブコマンド、FUSEによるマウント)

Singularityの仕様と実装 ー過去・現在・未来ー

過去
3.9以前

- OCI互換ランタイムは独自実装で、メンテナンス工数がかかる上、追従も困難だった。
- ホスト側の/etc/sub{u,g}idを用いた、ユーザー名前空間のみ対応。
- Encryption対応, OpenPGPによる電子署名など。
- GPU利用は/devを直接叩き、ランタイムライブラリはbindしていた。

現在
3.10 – 3.11

- OCIランタイム実装を削除し、OCI標準のruncを使うように変更。
- SBOM作成とセキュリティスキャンをSyft及びGrypeを用いて可能に。
- SquashFUSE, unprivileged cgroups v2 等に対応し、権限レス動作を向上。
- 電子署名の強化を検討開始。DSSEの実装等。
- NVIDIA GPUのリソース・ランタイム共有に関してnvidia-container-cli対応。

未来
4.0～

- SIF規格を拡張し、OCIイメージをSIFにカプセル化。OCIがSIFの恩恵を享受可能。
- 逆にSingularityがOCIのエコシステムをシームレスに利用可能に。
- remote/key コマンド群を見直し、個別実装されていたリポジトリ対応等を統合。
- GPUデバイスを/devの共有ではなく、CDI経由で行うように。

- Singularityの開発は、コミュニティ主導のCEを舞台に、ますます活発なチャレンジが行われている。その成果は数か月の検証とQAを経て、有償サポート付きのPROに反映されていく。
- Singularityの優位性は、利便性の高いランタイムとしてだけでなく、高パフォーマンスを実現可能なSIF形式によるところが大である。
- コンテナの標準規格であるOCIに対応することで、SIFの他ランタイムからの利用を可能とし、それに求められるセキュリティ対策について、趨勢に基づき順調に対応が進んでいる。
- 様々なコンテナイメージやランタイムが標準規格に準拠することで、シームレスに連携することが可能になると、容易に多様なワークロードをセキュアに展開できるようになると考えられる。



Thank you

sales@pacificteck.com